A geometric method for automatic generation of orthogonal grids on meander-like regions

Marco Paluszny, Gustavo Restrepo
mpalusznyk@unal.edu.co garestrear@unal.edu.co,
Accepted, January 2025

Abstract

The paper addresses the problem of finding an orthogonal mesh on a meander-like region, examples of these regions are riverbeds, coastal regions and in medical imaging, planar cross sections of bones (femur, humerus) or flattened curvilinear slices of veins and arteries described by developable surfaces. Essentially, it depends on two conformal maps $f(z) = \frac{1}{2}(z + 1/z)$ and $f(z) = (z + 1)(z - 1)$ that allow to transfer orthogonally gridded regions in an annulus to tiles that populate the meander-like region. The tiles are built using confocal lemniscates, ellipses and hyperbolas, which are stitched together to create an approximately orthogonal grid on the whole meander-like region which is adapted to both boundaries. The quality of the grid is assessed using standard criteria that involve angles and areas. As opposed to previous work we do not use sequential non linear optimization, instead we compare normalized segments of boundary with elements in a database. The database consists of approximately 50 thousand discretized arcs of lemniscates, ellipses, and hyperbolas. The computational efficiency of the comparison process is improved by clustering the database; this reduces the comparison to the fitting of normalized segments of the meander-like region with cluster representatives.

1 Introduction

This work presents a novel approach to constructing approximately orthogonal grids on $2D$ meander-like regions by combining conformal geometry and database search. A meander-like region is given by two sequences of points, or equivalently by two discretized curves that bound the region. In the case of a riverbed they are the two shores of the watercourse, other examples are isometrically unfolded curvilinear section of medical volumes. Rosenfeld, [10] gives the theoretical bases of the automation process of construction of such sequences of points.

Meander-like regions are an important area of investigation because they appear naturally in many fields: rivers in geography [13], regions along marine coasts [14], elongated organs in X-ray images or flattenings of curved slices in 3D medical data[1]. Our method is based on a multi-block technique, but introduces key innovations tailored to the unique characteristics of meander-like regions. Unlike traditional multi-block methods that often require interactive domain partitioning, our approach automatically divides the region, guided by its geometry. This automation significantly reduces preprocessing efforts and ensures that the blocks naturally conform to the irregular contours of the domain. It maintains grid continuity and orthogonality across block boundaries, avoiding misalignments or distortions that could otherwise compromise grid quality and simulation accuracy.

By incorporating these features, the proposed method combines the robustness of multi-block techniques with the flexibility required for meander-like geometries, making it particularly suited for such domains. Our algorithm is specifically designed to achieve orthogonality through the use of simple conformal maps. This technique involves mapping families of lemniscates and ellipses onto annuli of concentric circles. Additionally, the conformal technique used to build the grid simplifies any subsequent refinement or coarsening, which is particularly useful when dealing with extensive meander-like regions. Public domain general algorithms, such as Gmsh, do not prioritize grid orthogonality as the main objective.

Given the importance of good quality grids in the numerical solution of partial differential equations (PDEs), this topic has been explored systematically; [8] is an excellent reference. For this task, structured grids are preferred [2]. The classical reference for grids is [11]. See also [15] and [4]. In particular, orthogonal grids simplify the numerical solution of PDEs [3]. In other applications that involve measuring, they are useful for

[1] These might be extracted from medical volumes ([3]).

[2] Structured grids are those usually given by a repeating pattern such as a quadrilateral or a hexagon, guaranteeing the same number of edges meeting at every node.

[3] For example arising in modeling of contaminant dispersion in riverways.

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited

building approximately orthogonal coordinate systems adapted to the region's boundary which are useful for length measurements and volume estimation. The main limitation of the method (which is tied up to its main advantage: use of conformality as a unifying principle to create the grid) is that the sizes of cells may vary. Fortunately, the cell areas can be estimated using the conformality factor.; see [1].

Orthogonal or approximately orthogonal grids on meander-like regions provide coordinate systems adapted to the boundaries. The method to generate the grid is based on decomposing the meander-like region into subregions, which will be chosen to be conformally equivalent to tiles, carrying orthogonal grids. By stitching the gridded subregions together the whole meander-like region the full grid is built. The process is further optimized by clustering the database and refining the procedure by not comparing tiles given by segments between lemniscates, ellipses or hyperbolas but rather their one sided boundaries, with sides of subregions.

The paper is organized as follows: Section 2 introduces the problem and discusses previous work, Section 3 provides a short review of lemniscates, hyperbolas and ellipses as employed in the construction of orthogonal grids on meander-like regions, sections 4 and 5 look at the database, Section 6 considers implementation steps, Section 7 discusses the clustering of the database, Section 8 deals with postprocessing and Section 9 presents examples and discusses grid quality.

2 Previous work on meander-like region grids

Building orthogonal grids with lemniscates has been considered in [6], [7], [2] and [5]. The main idea is to fit segments of a meander-like region with segments of the area between two confocal lemniscates, ellipses or hyperbolas, Figure 1 shows an example using lemniscates with two foci. In particular [7] considers grids built with lemniscates of three foci, [6] looks at grid quality, [5] considers simplifying the problem by limiting the approximating lemniscates to two foci. This process relies on a sequence of non linear optimizations to determine the foci and radii of the lemniscates. In [2] the authors improve on the distance measurement between lemniscates of two foci and the points of the boundary of the meander-like region.

The improvements on previous work are:

- The need to fit pairs of boundaries of a meander-like region with pairs of confocal lemniscates regions using via nonlinear optimization is replaced by comparing normalized sequences of points.
- We further develop the technique of transferring the "polar coordinates" grid in an annulus without the need for lemniscates of more than two foci.
- Besides lemniscates, we now also use orthogonal families of ellipses and hyperbolas.

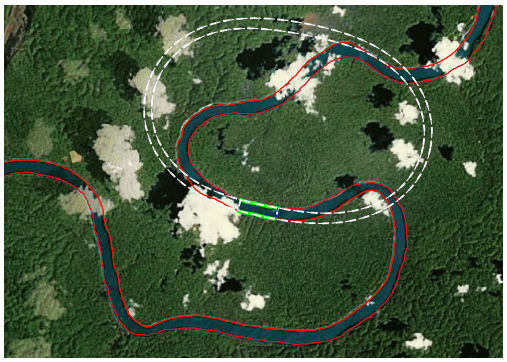


Figure 1: Confocal lemniscates fitting a meander-like region.

3 Lemniscates, ellipses, hyperbolas and grids

The region between two confocal lemniscates or two confocal ellipses carries an orthogonal grid, which is given by a sequence of intermediate corresponding curves and transversal hyperbolas, see Figure 2.

Figure 3 illustrates segments of the region between

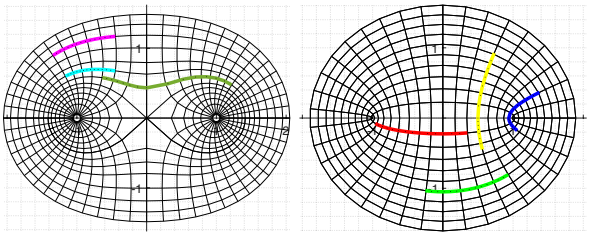


Figure 2: Families of lemniscates and ellipses.

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited

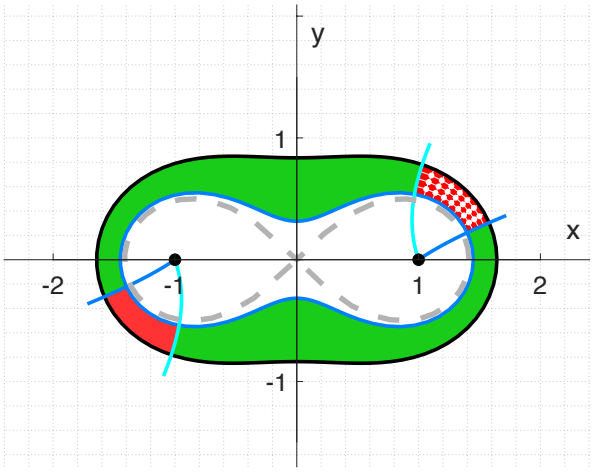


Figure 3: An orthogonal grid on a lemniscatic tile.

two confocal lemniscates, i.e. lemniscatic tiles. The orthogonal grid on a tile can be easily refined in the longitudinal and transversal directions. Our task is to fit subregions of the meander-like region with such orthogonally gridded tiles, as shown in red in Figure 3 (first quadrant).

An option is to choose subregions of the meander-like regions and fit them with orthogonally gridded segments "sandwiched" between ellipses, lemniscates or hyperbolas as shown in the previous Figure 1. Although this comparison process may be carried out, it is inefficient, because requires the simultaneous fitting of two confocal curves. A better alternative is to consider downward - upward concave sequences of consecutive points (as discussed in Section 4) of the boundaries and fit them with segments of lemniscates, hyperbolas or ellipses within a prescribed tolerance and - after this initial fit, proceed to choose the best tiles. The latter depends on the facing curves, i.e. for example if a segment of meander-like region is fitted with a segment of lemniscate then the orthogonally gridded tile is built with the corresponding segment of lemniscate, which fits best the opposite boundary of the meander-like region. This will allow for the construction of orthogonally gridded subregions on the meander-like region.

4 Concave segments

We consider discretized concave segments of lemniscates, ellipses, hyperbolas and also of boundaries of a meander-like regions. A segment that tends to bend in the same direction as we advance along it, will be referred to as a concave segment. The automatic division of a boundary of a long meander-like region into this kind of segments will be a step in the construction of the whole orthogonal grid.

For the sake of completeness we make this concept explicit: a sequence of points $z_j = x_j + iy_j, j = 1, \ldots, n$ is (upward or downward) concave if for any triplet of indices $j < k < l$ the points $x_k + iy_k$ lie on the same side of the straight line joining z_j and z_l within a given tolerance ϵ. More precisely, for example, the polyline with vertices $z_1, z_2, \cdots, z_n \in \mathbb{C}$ is downward concave within ϵ tolerance if

$$im\left((z_k - z_j)\cdot e^{-iArg(z_l - z_j)}\right) > -\epsilon, \tag{4.1}$$

for any $l = 3, 4, \cdots, n$, $j = 1, 2, \cdots, l-2$, $k = j+1, \cdots, l-1$.

Hence to verify that a segment of boundary of a meander-like region given by $z_1, z_2, \cdots, z_r$ is downward concave, it is enough to check for each pair $j < l$ and z_k such that $j < k < l$ that equation (4.1) is satisfied.

Concave segments of boundary of meander-like regions will be fitted with segments of lemniscates, hyperbolas and ellipses. For this we use a database and a normalization process. As suggested in Figure 2 we consider a full set of segments of lemniscates, ellipses and hyperbolas[4] We employ segments of these three types of curves because they cover most shapes that have to be approximated in order to tile a meander-like region. The elements of this database will be used to build the aforementioned tiles which will fit subregions of the meander-like region. Section 5 discusses in more detail the main features of the database. The tiles used to approximate segments of the meander-like region arise naturally from the elements of the database. Although most tiles will be bounded by segments of lemniscates and ellipses, the hyperbolas[5] will be useful in the approximation of "waist-like" subregions. In particular a discretized segment, as shown in Figure 2, together with any opposing segment[6] generate automatically an orthogonal grid on the tile determined by them. Hence

[4]The set of all the lemniscates of foci -1 and 1 and radii $\rho > 0$ will be referred to as a full set of lemniscates. The choice of maximum value of ρ guarantees that the corresponding lemniscate is essentially a circle, see ([9]). Similarly, for the full set of ellipses.

[5]We choose the segments of hyperbolas which are orthogonal to our family of ellipses.

[6]The opposing segment of any given segment lies on another

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited

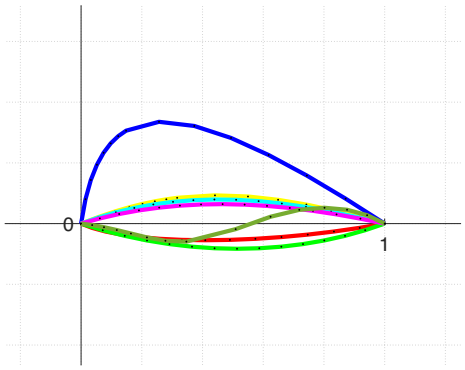


Figure 4: The normalized segments corresponding to the segments of Figure 2.

a given discretized segment yields a continuum of tiles: in the case of a segment of lemniscate, the opposing segment might correspond to any value of the radius ρ.

5 Database of segments

As mentioned above the database consists of discretized segments. They arise from continuous segments of lemniscates, ellipses and hyperbolas. In the case of lemniscates we consider the family of curves with foci in -1 and 1 (as described in Section 1), for ρ^2 discretized from 0.05 to 105, so that the density of ρ near 1 is higher than for values away of 1. The reason for the restriction of ρ to $[\sqrt{0.05}, \sqrt{105}]$ is that the lemniscate is essentially a circle for $\rho > \sqrt{105}$ and a pair of circles for $\rho < \sqrt{0.05}$. Indeed, as $\rho \to \infty$ the lemniscate tends to a circle, and in fact if $\rho = \sqrt{105}$ the lemniscate coincides with a circle of radius $\sqrt{105}$ within a 5% deviation in the curvature, see [12].

A lemniscate segment is a connected component of the inverse image of an arc of circle of the map, given in terms of complex numbers, by $f(z) = z^2 - 1$. We consider approximately 750 segments per lemniscate, each segment is normalized (see [4]) to join 0 and 1, to include in the database. Further, each normalized segment is discretized, trying to achieve a balance between the cost of computing distances between normalized segments of the meander-like region and normalized lemniscate segments, within a fitting accuracy. For, extensive experimentation shows that the uniform discretization of the corresponding arc of circle into 15 points is a reasonable choice[7]

Similarly, a convenient number of ellipses is 15 and on each half ellipse we consider the images of 30 uniformly distributed points on a half circle under the map[8]

$$f(z) = \frac{1}{2}(z + 1/z) \tag{5.2}$$

As for lemniscates the segments of ellipse have endpoints at these image points, they are normalized and discretized similarly.

The segments of hyperbolas are the images of segments of rays of circles under the map (5.2). We use the 15 hyperbolas (two branches each) resulting from the 30 point partition of the ellipses. Each segment of hyperbola is normalized and is described by a sequence of 15 points.

Summarizing, the database consists of sequences of 15 points in $2D$ which arise from the discretization of normalized segments of lemniscates, ellipses and hyperbolas.

As built, the database contains discretized segments which may be similar. This fact is advantageous because the database is used to find tiles which fit subregions of the meander-like region and, similar or even equal segments may be associated to very different tiles. Figure 5 illustrates two examples of similar segments of lemniscates that produce very different tiles.

We chose these families of curves: lemniscates, ellipses and hyperbolas, because the region between any pair of lemniscates or any pair of ellipses is conformally equivalent to an annulus i.e., the region bounded by two concentric circles which carries naturally an orthogonal grid.

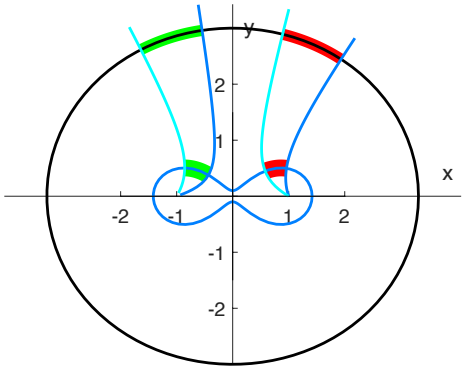
To build the orthogonal grid, the fitting process is actually a comparison procedure of normalized segments of boundary with elements of the database. The detailed description is given in Section 8. In Section 7 we show how this comparison process actually does not require to look at all the elements of the database.

curve of the same family (namely, lemniscate, ellipse or hyperbola) which is bounded by the same orthogonal curves to this family, as the given segment.

[7] Besides keeping low the cost of distance computation, the chosen discretization is also adequate for k-means calculations and storage.

[8] The map f, for convenience, is expressed in terms of complex numbers. It takes points in $2D$ into points in $2D$, it takes circles centered at the origin into ellipses.

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited



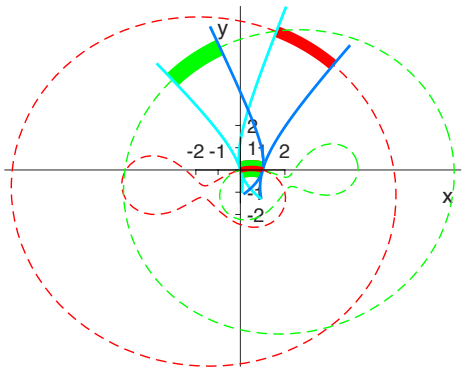


Figure 5: Equal segments that produce different tiles, because they lie in different zones of the same lemniscate or lie on different lemniscates.

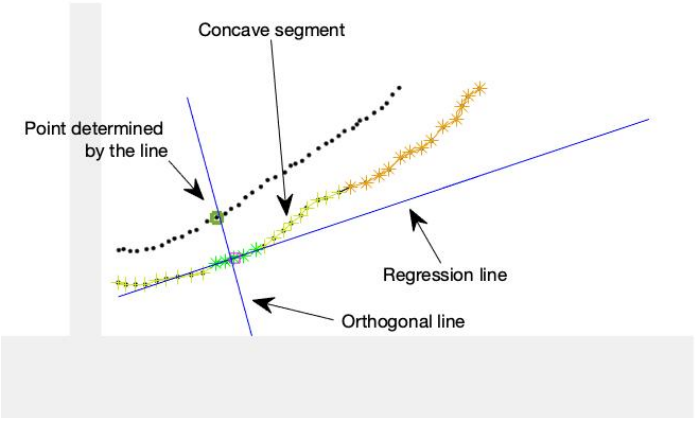


Figure 6: Light green and orange sequences of points are discretizations of consecutive concave segments of a boundary. The lower red square is the point corresponding to the middle index. The perpendicular foot to the regression line at this point determines the nearest point of the opposite boundary discretization.

6 Implementation

Our methodology proposes comparing elements in the database with segments of boundary of the meander-like region. Hence, attention needs to be paid to the choice of these boundary segments: if the given segment is concave the opposite segment should also be concave. This imposes restrictions on the choice of the segments of boundary, facilitating the construction of the grid. As shown in Figure 6 for each concave segment of boundary, we compute the regression line at the midpoint of its discretization and find the nearest point on its perpendicular foot to the opposite boundary. From the latter we grow a maximal sequence of points on the opposite boundary (the same number in each direction) which satisfies the concavity condition. For downward concavity the condition is given explicitly by 4.1 of Section 4.

The goal is to fit a gridded tile that fills maximally the area between these two segments. The detailed process is as follows. For each such segment which has been approximated with an element of the database and its facing segment of the opposite boundary, we find the arc of the family of curves of the fitting element that best approximates the latter. For example, for a segment of the meander-like region whose approximating arc is a segment of lemniscate, its opposing boundary is fitted with a confocal lemniscate. This is illustrated in Figures 7, 8, 9 and 10.

- Figure 7 shows a subregion consisting of a concave segment (green) and an opposing segment of the

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited

meander-like region (red) as they lie in the physical meander-like region.

- Figure 8 shows both segments after the normalization of the concave green segment (which now starts at 0 and ends at 1) and the corresponding transformation of the red segment. Note the need for trimming, in order to choose the tile. It also highlights the best approximating element in the database (black dotted) of the green segment and the lemniscate (adjusted by normalization) on which it lies is drawn for context.

- Figure 9 displays both normalized segments of boundary (green and red) in lemniscate space, i.e. the green segment is shown to lie on a lemniscate (dotted black) on which it chooses a segment. The red segment lies on a confocal lemniscate. The black dotted, vertically pointing segments at the extremes of the green segment, are arcs of hyperbolas orthogonal to the family of lemniscates[9], see [6].

- Figure 10 exhibits the circle corresponding to the lemniscate in Figure 9 and a circle whose corresponding lemniscate approximates the red segment of Figure 9. The vertical leaning hyperbolas in this figure are mapped to the radii under the map $f(z) = z^2 - 1$.

- Figure 11 illustrates the trimming of both segments (in circumference space) bounding them with two rays.

- In Figure 12 the trimmed region in circumference space is taken back to a region in lemniscate space which inherits the orthogonal grid from the annular region in Figure 11 by conformality.

- In Figure 13 a normalized segment of boundary of the meander-like region (i.e. the green curve segment joining 0 and 1) is fitted with a lemniscate element of the database together with the transformed red segment of the meander corresponding to a confocal lemniscate. Both lemniscates in light blue are drawn for context.

- Figure 14 displays the tile resulting from the starting concave segment (green) and an opposing segment of the meander-like region (red). The lemniscates, under the transformations to physical space, are drawn for context. It also shows the need of further cropping of the lower boundary of the subregion in order to fit a tile, as suggested in Figure 11. Figure 15 illustrates the need of trimming of both segments of the boundary of the meander like region to be fitted by a lemniscate depicted in solid green. The cropping introduces gaps that are dealt with in Section 8. An alternative, to minimize the number of gaps is to allow boundaries to have intersections.

The concave segments of the meander-like region can be chosen without any prescribed order[10]. The algorithm may also be applied to more complex meander-like regions, by splitting the region into simpler ones.

Figure 7: Physical region.

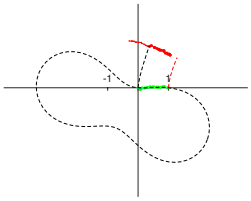


Figure 8: Normalization.

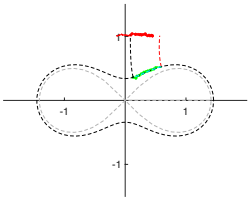


Figure 9: Lemniscate space

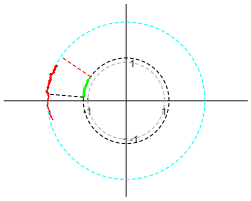


Figure 10: Circle space.

7 Metrics and clusterization

After normalization, the fitting of the boundary of a subregion of a meander-like region involves the distance to the discretized segments of the database. We use the standard Euclidean metric between sequences of points

[9] The family of confocal lemniscates with foci -1 and 1 are conformally equivalent $(2:1)$ to the family of circles centered at the origin, under the map $f(z) = z^2 - 1$.

[10] Simultaneous fitting of various segments allows for faster processing, which is especially useful in the case of very long rivers.

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited

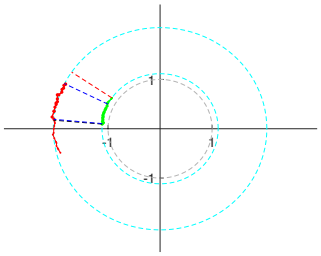


Figure 11: Trimmed circle region.

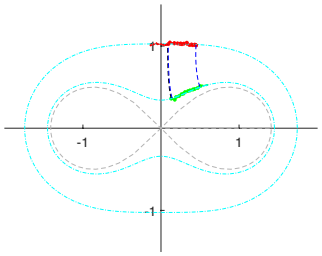


Figure 12: Back to lemniscate space.

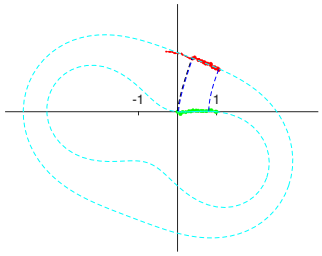


Figure 13: Back to normalization space.

Figure 14: The fitting tile in physical space.

[11]. Since the database contains 53179 elements and for each boundary segment we need to find the best fit, a computationally expensive process, we introduce a strategy to lower this cost. Namely, we organize the database in clusters of similar elements and compare each normalized boundary segment with a representative of each cluster and then choose the nearest element of the cluster.

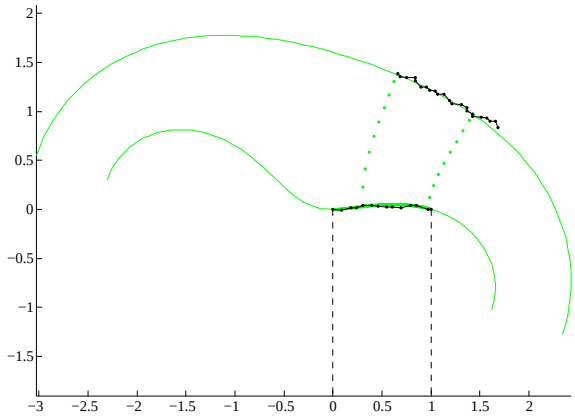


Figure 15: Trimming with two orthogonal hyperbolas.

We clusterize the database using k-means with Euclidean distance. To determine the number of clusters we employ the "elbow method" which consists in looking at the graph of the number of clusters vs a cost function that involves the distance of each point to the centroid of its cluster, for an intuitive presentation of the method). In our case the resulting number of clusters for the database of 53179 elements is 510.

Given a segment of boundary of the meander-like region, the above clusterization allows for a more efficient choice process of fitting element in the database: compare the boundary segment with each cluster centroid and choose the best fit among the elements of the cluster. Since the normalized segments of boundary belonging to a cluster might not have the same number of points we use the Hausdorff distance for within comparison in each cluster. In Section 9 we compare CPU times for the building of the grid with and without clustering, for specific examples.

8 Postprocessing: gaps and intersections

As presented in Figure 6, once the opposing boundaries have been chosen they are fitted with two curves of the same family, namely two confocal lemniscates, ellipses or two hyperbolas. In order to find a tile we need to trim them with orthogonal curves. The latter are chosen to

[11] A more precise distance measurement is the Hausdorff discrete distance but the k-means, but the clustering process for our database of discretized segments (i.e. polylines consisting of 15 points joining $(0,0)$ and $(1,0)$) is computationally too expensive.

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited

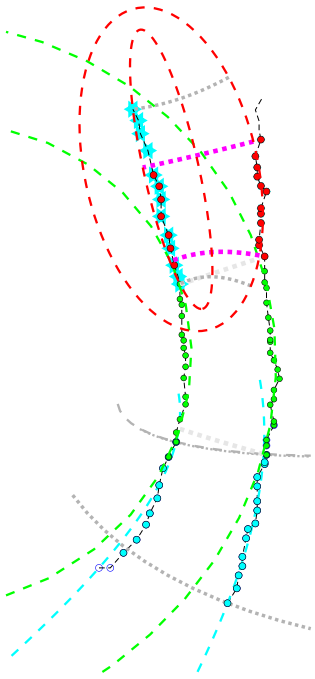


Figure 16: Blue and red point sequences of the meander-like region are fitted by ellipses and trimmed by magenta hyperbolas.

be those that connect best the boundaries.

An example of this process is illustrated in Figure 16 where the upper red points and the lower blue starred points are fitted with two confocal ellipses and are cut by four orthogonal hyperbolas determined by the endpoints of the red and blue starred points. The two hyperbolas that will bound the leftmost tile are shown in magenta.

In Figure 17 we overlay three consecutive tiles with their orthogonal grids as generated by the trimming curves. This figure also shows gaps between consecutive tiles introduced by the trimming. The gaps, as well as intersections between tiles, are allowed to occur in order to increase the odds of finding segments of boundaries coupled with opposing segments of the meander-like region, as shown in the construction illustrated in Figure 6. Later, these may be fitted with curves of the same family, and hence may be gridded.

Figure 18 shows gaps and intersections between consecutive fitting tiles along a larger portion of a meander-like region.

In order to extend the grid to cover all the gaps, which are depicted in Figure 18, we proceed as follows. In case of a gap between two neighboring tiles we remove some grid points on both of its sides. In Figure 19 they are given by orthogonal hyperbolas and are identified by red squares. This expands the gridded gap to a

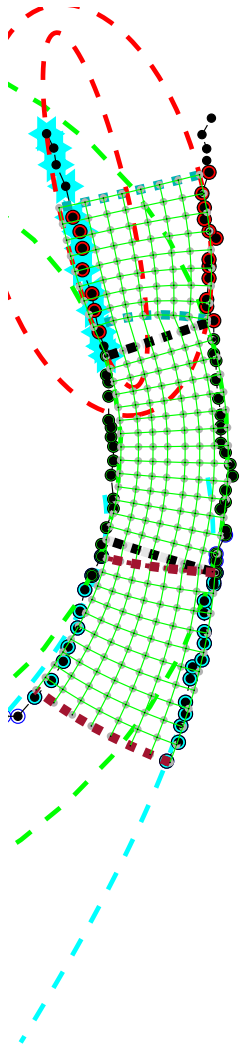


Figure 17: Orthogonal grids on three consecutive tiles as exemplified in Figure 16. Each tile is bounded by two curves of the same family and two orthogonal trimming curves. For example the middle tile is given by two confocal lemniscates (green dashed lines) and the construction of its orthogonal grid is illustrated in Figure 3.

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited

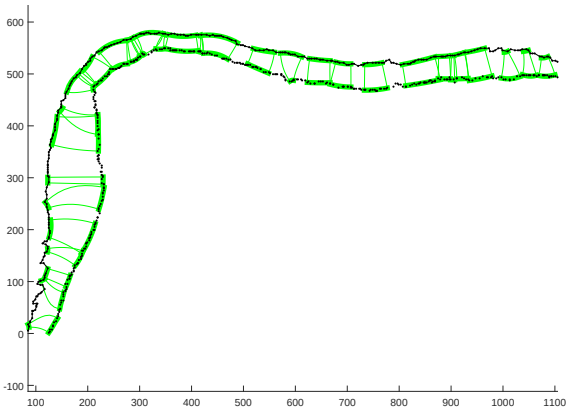


Figure 18: Detecting segments of the meander-like region that correspond to tiles. The sector near (200,400) shows an intersection, and near (800,500) and (600,500) we have gaps.

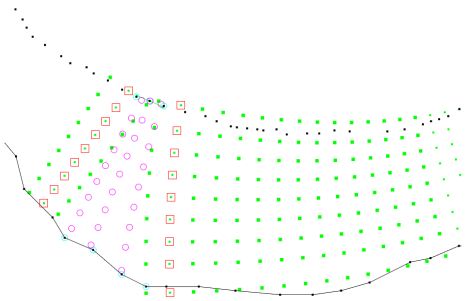


Figure 19: The triangle-like region between the tiles is a gap in the grid. The grids of both tiles are given by green points. In order to extend the grid to the missing gap the grids of the consecutive tiles are trimmed by removing the grid points that bound the triangle, so the old tiles are replaced by two new tiles and the red boxed points are the new bounding hyperbolas. The larger empty region is gridded by applying again the algorithm presented in Section 6. The magenta circles are the new grid points of the gap.

trapezoid-like subregion containing the gap. To build a grid on that trapezoid we proceed as before searching the database to match one of its boundaries and finding a curve of the same family that approximates the other boundary. If the trapezoid does not have enough points on each boundary side, namely at least 4 sample points, the grid is built using Bézier curves that are approximately orthogonal to both boundaries.

When neighboring tiles intersect then by construction this intersection involves only one bounding curve of each tile. So we force a gap by removing the offending curves and proceed as above.

9 Example and grid quality

Our first example is an orthogonal grid along the coastal ribbon of San Andres Island, extending approximately 13 km. The discretizations of the inner and outer boundaries of this meander-like region consist of 1102 and 1346 points, respectively.

The grid quality parameters are as follows: average aspect ratio (AAR): 1.33, maximum aspect ratio (MAR): 3.37, percentage of tiles with aspect greater than 1.3: 2.75, average deviation of ortogonality (ADO): 0.11, maximum deviation of orthognality (MDO): 1.49 and percentage of tiles with deviation of orthogonality (PDO) greater than 1: 0.18.

The second example is the extension of the Apaporis river that joins Pacoa (Vaupés, Colombia) and Vila Bittencourt (Amazonas, Brasil), its length is 396 km, the numbers of sample points on each boundary are approximately 11.500 and 13.000. It was run on a Windows machine, Dell Latitude 5480, 16 Gb RAM, Intel Core i7, 2.80 GHz CPU. In this example the fitting process takes 61 minutes without clustering and under 20 minutes with clustering. In gap filling there is no difference.

Figure 20 illustrates some representative pieces of the full 396 km Apaporis grid.

It is well known that the automatic generation of high-quality orthogonal grids is costly. However, the clustering feature significantly improves the overall process. In the Apaporis example, it is over 29% faster compared to when no clustering is used and the grid quality parameters are as follows: without clustering AAR is 1.22, MAR is 24.67, percentage of tiles with aspect greater than 2: 1.70, ADO is 0.04, MDO is 1.55 and PDO greater than 1: 0.12. With clustering these numbers are slightly better.

10 Conclusions

We presented an algorithm to build orthogonal grids on meander-like regions. It is based on the comparison of normalized boundaries of subregions with a database of segments of lemniscates of two foci and families of ellipses and hyperbolas. The process consists of the following steps:

- Determine subregions of the meander-like region

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited

Figure 20: Segments of Apaporis river grid.

of similar or opposite concavity (within the given concavity-tolerance), which face each other.

- Normalize one of the boundaries of the subregion, so that its endpoints coincide with 0 and 1 and find the fitting segments within the approximation tolerance. The fitting tolerance is derived from the mean distance between consecutive points of the boundaries.
- Each fitting segment points to one curve of our curve families: lemniscates, ellipses or hyperbolas; they define tiles, i.e. regions bounded by two curves of the same family and pairs of transversal orthogonal curves. This allows for the immediate building of orthogonal grids on such subregions.
- The procedure generates gaps which are gridded after taking into account all the regions approximated by lemniscatic and elliptic sectors.

Future work with the techniques developed in this paper includes enhancing the methods to identify predominantly concave segments of the boundaries of meander-like regions, potentially utilizing AI techniques. Additionally, for our easily refinable and adaptable orthogonal grids, it seems reasonable to consider the conformality factor as a means to estimate size, which will enhance their usefulness as coordinate systems.

In the additional materials the interested reader will find all the Matlab scripts, the database of segments of lemniscates, ellipses and hyperbolas, the full Apaporis example as well as other examples together with their underlying images.

Acknowledgments

M. Paluszny would like to thank Universidad Nacional de Colombia and MinCiencias for support through project GeoMed.

References

[1] J U Brackbill. Coordinate system control: adaptive meshes.

[2] L. G. Cardona, M. Lentini, and M. Paluszny. Grid generation using lemniscates with two foci. *Mathematical and Computer Modelling*, 2186–2195, 2013.

[3] Steven Haker, S. Angenent, and Ron Kikinis. Nondistorting flattening maps and the 3d visualization of colon ct images. 03 2001.

[4] Patrick Knupp and Stanly Steinberg. *Fundamentals of grid generation*. CRC press, 2020.

[5] M. Lentini and M. Paluszny. On grid generation with lemniscates of two and three foci. *Computational and Applied Mathematics*, 32(3):495–507, 2013.

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited

[6] Marianela Lentini and Marco Paluszny. Approximation of meander-like regions by paths of lemniscatic sectors and the computation of orthogonal meshes. *Computer Aided Geometric Design*, 25:729–737, 2008.

[7] Marianela Lentini and Marco Paluszny. Orthogonal grids on meander like regions. *ETNA Electronic Transactions on Numerical Analysis*, 34:1–13, 2008.

[8] Vladimir D. Liseikin. *Grid generation methods*. Springer, 2010.

[9] G. Restrepo. *Método geométrico para la generación de mallas ortogonles sobre regiones tipo meandro*. PhD thesis, Escuela de Matemáticas. Facultad de Ciencias. Universidad Nacional de Colombia, May 2022.

[10] A. Rosenfeld and A. C. Kak. *Digital Picture Processing*. Academic Press, Inc. Orlando, FL, USA., 2nd edition, 1982.

[11] Joe F Thompson and Nigel P Weatherill. Structured and unstructured grid generation. In *High-Performance Computing in Biomedical Research*, pages 63–111. CRC Press, 2020.

[12] S. Verblunsky. On the curvature of level curves. *Mathematische Annalen*, 125(1):472–476, 1952.

[13] F. Yang, X. Shao, X. Fu, and E. Kazemi. Simulated flow velocity structure in meandering channels: Stratification and inertia effects caused by suspended sediment. *Water (Switzerland)*, 11(6), 2019.

[14] Yaoxin Zhang, Yafei Jia, Mustafa Altinakar, Yan Ding, Vijay Ramalingam, and Soumendra Kuiry. Structured mesh generation along louisiana- mississippi coastline for simulation of coastal processes. In *10th Intl. Conf. on Hydroscience and Engineering. Orlando, Florida, U.S.A. Nov 4-7, 2012.*

[15] Yongjie Jessica Zhang. *Geometric modeling and mesh generation from scanned images*. Chapman and Hall/CRC, 2018.

Copyright © 2025 by SIAM
Unauthorized reproduction of this article is prohibited